# Game Theoretic Multiple Mobile Sensor Management under Adversarial Environments

**Mo Wei**
DCM Research Resources, LLC
Germantown, MD, U.S.A.
superninheadbird@yahoo.com

**Genshe Chen**
DCM Research Resources, LLC
Germantown, MD, USA
gchen@dcmresearchresources.com

**Erik Blasch**
AFRL, WPAFB
OH, U.S.A.
erik.blasch@wpafb.af.mil

**Huimin Chen**
Depart. of Electrical Engineering
University of New Orleans
New Orleans, LA, U.S.A.
hchen2@uno.edu

**Jose B. Cruz, Jr.**
Department of ECE
Ohio State University
OH, U.S.A.
cruz@ece.osu.edu

***Abstract - This paper presents a game theoretic approach for the management of multiple mobile sensors. Our approach can maintain tracks of smart targets under possibly adversarial environments. To ensure computational tractability, sensor management is divided into sensor assignment and sensor scheduling. In sensor assignment, covariance control and information theoretic sensor assignment are combined logically. In sensor scheduling, the targets are modeled as entities with different levels of intelligence, which will invoke different strategies of sensor scheduling. Simulation results demonstrate the effectiveness of the proposed sensor management scheme.***

**Keywords:** Sensor management, game theory, covariance control, information theoretic sensor assignment.

## 1 Introduction

When using multiple sensors in an automatic target recognition (ATR) and tracking system, efficient sensor management strategy plays an important role in achieving high performance for the overall system. According to [1], sensor management can be treated as a general strategy that controls sensing actions, including generating, prioritizing, and scheduling sensor selections and actions. Specifically, sensing actions include but are not limited to illuminating a target, selecting sensing mode, scanning an area for new targets, etc. Usually, the input to the sensor management system can be a target state estimate and the corresponding error covariance from the tracking module as well as target features/IDs from the ATR module. The output of the sensor management system can be sensor-target assignment and schedule of sensing actions in the future.

Usually, sensor management has to deal with two important topics, namely, *sensor assignment* and *sensor scheduling*, although they are often tightly coupled. Sensor assignment decides which sensor or sensor combination will be assigned to which target or which area. Sensor scheduling determines when and which sensor will take what action (e.g., pointing to which target or which area). In other words, sensor assignment mainly deals with the issues over sensor/target/environment/space horizon, while sensor scheduling mainly determines the sensing actions over the time horizon. In real world applications, sensor assignment and sensor scheduling are often optimized jointly to help improve ATR and tracking performance, reduce the requirement of system resources, and even reduce risks in the context of persistent surveillance. Likewise, sensor management, based on either predictions or cost minimization functions, ensures that the right sensor is activated to illuminate the target of interest for a given spatial/spectral environmental condition. Knowing all possible scenarios is difficult to do *a priori*, so care must be taken in tradeoffs between (1) online versus a off-line (i.e. models) analysis, (2) metrics for arbitrating between sensor selections, and (3) search versus evidence maintenance.

The following input-related issues should be considered when designing a sensor management module.

I1) number of sensors and sensor information, such as types, ranges, modes, capacity, etc.

I2) number of targets and target information, such as types, related tracks, whether target being intelligent in its behavior modes, etc.

I3) terrain , weather, and illumination conditions.

I4) physical constraints such as energy limit, operation time limit, communication constraints, bandwidth, etc.

I5) user requirements such as computational load, centralized/decentralized configurations, detection probabilities, false alarm probabilities, tracking/classification accuracies, risks, etc.

Possible outputs of a sensor management module include decisions on

O1) which sensor (combination) is assigned to which target (combination) or which area at which period

# Report Documentation Page

*Form Approved*
*OMB No. 0704-0188*

Public reporting burden for the collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to a penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number.

| | | |
|---|---|---|
| 1. REPORT DATE<br>**JUL 2008** | 2. REPORT TYPE | 3. DATES COVERED<br>**00-00-2008 to 00-00-2008** |
| 4. TITLE AND SUBTITLE<br>**Game Theoretic Multiple Mobile Sensor Management under Adversarial Environments** | | 5a. CONTRACT NUMBER<br>5b. GRANT NUMBER<br>5c. PROGRAM ELEMENT NUMBER |
| 6. AUTHOR(S) | | 5d. PROJECT NUMBER<br>5e. TASK NUMBER<br>5f. WORK UNIT NUMBER |
| 7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)<br>**Air Force Research Laboratory,Wright Patterson AFB,OH,45433** | | 8. PERFORMING ORGANIZATION REPORT NUMBER |
| 9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES) | | 10. SPONSOR/MONITOR’S ACRONYM(S)<br>11. SPONSOR/MONITOR’S REPORT NUMBER(S) |

12. DISTRIBUTION/AVAILABILITY STATEMENT
**Approved for public release; distribution unlimited**

13. SUPPLEMENTARY NOTES
**11th International Conference on Information Fusion, June 30 ? July 3, 2008, Cologne, Germany.**

14. ABSTRACT
**see report**

15. SUBJECT TERMS

| 16. SECURITY CLASSIFICATION OF: | | | 17. LIMITATION OF ABSTRACT | 18. NUMBER OF PAGES | 19a. NAME OF RESPONSIBLE PERSON |
|---|---|---|---|---|---|
| a. REPORT | b. ABSTRACT | c. THIS PAGE | | | |
| **unclassified** | **unclassified** | **unclassified** | **Same as Report (SAR)** | **8** | |

**Standard Form 298 (Rev. 8-98)**
Prescribed by ANSI Std Z39-18

O2) which sensor should be in which mode, at what revisit time period, and with what signals being emitted, etc.

O3) which sensor should illuminate which direction and/or move in which direction (if mobile).

O4) which exploitation or classifier (or tracking) algorithm to invoke for a given condition.

O5) what fusion information metrics should be delivered to the user.

It should be noted that since considering so many issues (I1~I5 and O1~O5) simultaneously is too difficult, real world applications of sensor management algorithms can only address some of the issues listed above. Furthermore, the complexity of the problem space requires intelligent strategies to focus the sensor management requiring a host of metrics to afford effective optimization.

Most of the existing sensor assignment algorithms try to select sensors/targets so that a performance metric is optimized [1]. Such a performance metric, i.e., the objective function, can be the trace of each target's state estimation error covariance weighted by target importance [2], the information gain [3], or some objective based on the covariance control [4]. Currently, the most popularly used sensor assignment algorithms are either based on the information theoretic approach [3] or the covariance control approach [4].

The *Information theoretic approach* tries to maximize the information gain [1], which is a data-independent indicator of the usefulness of obtaining the target information through one observation at time *k* defined by

$$I(P(k\mid k-1), P_i\ (k\mid k)) = \frac{1}{2}\ln\frac{\mid P(k\mid k-1)\mid}{\mid P_i\ (k\mid k)\mid} \qquad (1)$$

where $I(\cdot,\cdot)$ is the information gain in the Fisher sense, $P(k\mid k-1)$ is the prior error covariance of the target track, and $P_i\ (k\mid k)$ is the posterior covariance after applying the estimate of the $i$-th sensor combination. Usually, the sensor combination which can achieve the maximum information is assigned to this target or a combination of targets.

*Covariance control method* starts with the goal related to the estimation of error covariance which can be determined by specific mission requirements such as the desired covariance to locate an enemy target before launching a rocket. Then one seeks to optimize specific covariance related objective function such as the eigenvalue/minimum goal [4]:

$$\begin{aligned} \Phi_{ev} &= \left\{\Phi_i : P_d - P_i\ > 0\right\} \\ i_{ev} &= \arg\min_i \mid\Phi_i\mid,\ \ \Phi_i \in \Phi_{ev} \end{aligned} \qquad (2)$$

where $\left|\Phi_i\right|$ is the number of sensors in $i$-th sensor combination, $P_d$ is the desired covariance, and $P_i$ is the covariance provided by $i$-th sensor combination. Since for a multi-sensor multi-target system a whole binominal combination search will require a computational load on the order of $o(2^{N_s} N_t)$, where $N_s$ is the number of the sensors and $N_t$ is the number of targets, greedy algorithms (or "myopic") are often applied to reduce the computational load to $O(N_s^2 N_t)$.

In general, information based approaches try to maximize the utility of available sensors, while covariance control approaches try to meet specific goals with minimum sensor resources such as sensor numbers. As stated in a research about comparison between these two approaches [4], when there are many more sensors than targets, information based approaches work better. In contrast, covariance control based approaches work better when there are relative fewer sensors. To find an efficient algorithm for unknown or time varying number of targets is still an open problem and solutions are often scenario-specific.

Sensor scheduling often relies on advanced optimization techniques such as dynamic programming [5] and Q-learning [6], which is often applied to provide approximate values. A nonlinear particle filter method [7] is also frequently applied to target state estimation with nonlinear system dynamics. It can be combined with Q-learning, to generate various hypotheses over one look-ahead horizon. Theoretically speaking, longer look-ahead horizon implies "non-myopic" and can provide better performance over the long run. However, when the look-ahead horizon is too far, it will have to rely on too many predicted covariance's or information gains thus being sensitive to modeling error. In addition, an overly stretched look-ahead horizon often implies unaffordable computational load. As a result, one has to carefully choose a reasonable time horizon for the problem at hand to avoid degradation of overall performance.

Currently, sensor assignment and scheduling methods have been extensively studied and the field becomes relatively mature. Many researches have focused on fusing more knowledge (such as target motion modes and road network topology) as well as designing specific performance metrics (such as target/sensor valuation) and determining the appropriate criteria (such as horizon length and hypothesis determination thresholds) more suitable to specific applications with various practical constraints (such as communication capacity/delay and terrain conditions) [8][9]. Some researchers also introduced cooperative game theory to help improve performance under decentralized situation [10][11]. Most of these efforts are confined to different practical applications and greatly contribute to the ATR and tracking research. However, two issues concern "intelligent" targets and tradeoffs between fusion performance and system requirements.

Generally speaking, the approaches discussed above work well under traditional non-intelligent ATR and tracking environments in which there are no "intelligent targets". Here an *intelligent target* (also called a "smart target") is a target that can be aware of or even rationalize whether it has been detected/tracked or will be detected/tracked, and can engage in launch some specific actions accordingly to prevent the sensors from accurately detecting/tracking it. Recently, more and more targets with intelligent behaviors are emerging in ATR and tracking area research, such as automobile drivers with counter-speed radar, enemy tanks with radar wave detectors, etc. Tracking such targets often requires rational analysis on both sides using non-cooperative game theory [11]. Moreover, sometimes such smart targets might use random strategies in their actions. For example, although a target knows that it has been tracked, it might not always choose the best action obtained using game theoretic analysis, say choose the best action with probability 0.5 and stay dumb otherwise. This will make the prediction using purely game theoretic approach faces additional difficulty in modeling the rationality of the opponent.

The second issue of sensor management in modern tracking applications is the tradeoff between different *performance metrics and system requirements*. For example, for a practical tracking system to monitor smart targets, many conflicting interests need to be considered: the competition between track maintenance performance and computational load, the tradeoff between short term accuracy and long term track continuity, etc. In addition, a practical sensor management module should not be too complex, no matter what is implied as theoretically optimum, implementation complete, or operational robust. As a result, full horizon search for the best strategy is often infeasible. For computational efficiency, some suboptimal (e.g. approximation) approaches have to be applied for sensor resource management. However, different suboptimal approaches often have different strength and weakness thus only suitable for specific applications. For example, the information theoretic approach (usually applied with pure greedy algorithm) might cause some targets to starve[1] while other targets are covered with more sensors than necessary. A pure covariance control approach (usually applied with need-based greedy algorithm [12]) goes to the other extreme and can save sensor resources when there are relatively limited sensor resources, but often performs worse than pure greedy algorithm when there are more than adequate sensor resources. An algorithm that can inherit the strength and avoid the weakness of both methods is desired.

# 2 Hierarchical Sensor Management

We propose a hierarchical sensor management (HSM) scheme for both sensor assignment and sensor scheduling to monitor smart targets. Sensor management will assign sensors to specific targets or areas, and sensor scheduling will schedule the actions (including sensor motion if the sensor is mobile) for each sensor. HSM integrates both the information theoretic approach and covariance control based method so that the system can perform well in environments the changing number of targets and tracking performance requirements. For sensor scheduling, we consider both the cases with ideal rationality on both sides and the cases in which smart targets act with some randomized strategy. A learning mechanism, in conjunction with possible classification knowledge and game theoretic calculation, will be used to automatically identify whether the target responds with randomized strategy and if yes, what is the extent of randomness. The system is suitable for managing heterogeneous sensor networks including airborne, space based, ground based and sea based EO/IR/radar sensors [13] with possible terrain constraints. An illustrative scenario of a sensor management system with networked heterogeneous sensors is shown as follows (Figure 1).

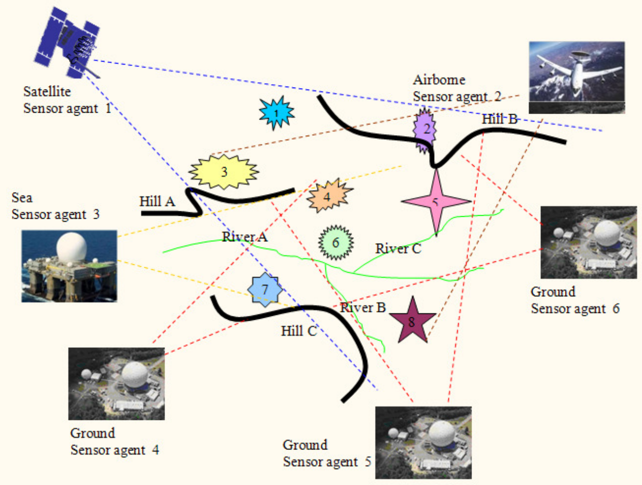


Figure 1: An illustrative sensor management scenario.

## 2.1 Sensor Assignment

The basic idea of our HSM sensor assignment approach is a two step procedure: We first apply covariance control, then switch to information based algorithm after all existing covariance requirements are satisfied. In this way we can largely take the advantages but avoid shortcomings of both methods simultaneously.

To understand the underlying philosophy, we first provide the basic logics of the two existing sensor assignment algorithms. A simple description of covariance control algorithms' logic is: treat the targets as "customers" with *explicit and fixed needs* and do what we need with the least amount of resources. This is because even if we can maximize the total information gain, but the needs of customers can not be completely satisfied, this assignment is still not a good solution. On the other hand, information theoretic approaches tend to treat the target (or sensors) as "customers" with *inexplicit needs* and try to

[1] Here starve means no sensor resource is assigned to supervise such targets.

maximize the information gain, which is assumed to be the unique need of all customers. As a result, when there are still explicit requirements unsatisfied, covariance control is a good choice. When all explicit requirements have already been satisfied and there are still sensing resources available, information theoretic assignment is more appropriate. In this way, both approaches can be integrated and the transition is naturally linked to supply and demand analysis. The computational load will be compatible with the two methods thus for sure tractable. We describe the HSM algorithm next.

For a scenario with $N_s$ heterogeneous sensors and $N_t$ targets[2], our sensor assignment algorithm is summarized as follows:

***Step 1:*** If there is no explicit target requirements, go to step 6. Else, calculate the "needs" $n(t)$ for target $i$'s according to

$$n(i) = -\min\{\text{eig}[P_d - P_i\ (k \mid k-1)]\}(10 - i_p) \qquad (3)$$

where $i_p$ is the priority of target $i$. Note that the number $10$ in $(10 - i_p)$ is an example recommended by [4] for general cases and can be adjusted according to specific applications. For example, if there are only three different priorities, we may set it to 4. The reason that we do not set it to 3 is to avoid such need being zero. To replace $(10 - i_p)$ directly with target importance is also a feasible approach. Equation (3)'s idea is to meet the desired covariance along the axis corresponding to the worst case difference in eigenvalue. In this way, more information about the error covariance can be utilized than in using trace or determinant [2].

***Step 2:*** Select the target with the largest "need" as the target we will consider.

***Step 3:*** Calculate the updated a posterior covariance resulted from using a sensor $j$ according to

$$P_i\ (k \mid k, j) = (I - K_i\ (k, j) H(j)) P_i\ (k \mid k-1) \qquad (4)$$

The covariance for covering an area can be calculated according to the strategy in [3][14].

***Step 4:*** Assign the sensor that maximizes

$$\min\{\text{eig}[P_d\ (t) - P_i\ (k \mid k, j)]\} \qquad (5)$$

to target $i$.

***Step 5:*** Do the following updates

$$P_i\ (k \mid k, j) \to P_i\ (k \mid k-1)$$

$$-\min\{\text{eig}[P_d\ (t) - P_i\ (k \mid k, j)]\}(10 - i_p) \to n(i)$$

then go to ***Step 1***.

***Step 6:*** If there are no available sensors, go to ***Step 8***. Otherwise, for all available sensors in available sensor set $S$, calculate reward to sensor set $j$

$$r_j(S_j, S_{-j}) = \alpha_j + (1 - \alpha_j)\left[1 - \exp\left(-\alpha_j \left|S_j\right|^{-1} \sum\nolimits_{k \in S_j} \eta_k h_{ij}^T R_{ij}^{-1} h_{ij}\right)\right] \qquad (6)$$

where $\alpha_j \in [0,1]$, $\left|S_j\right|$ is the size of the target subset, $\eta_k$ is the target information weight determined by the target importance and anomaly levels fed back from last calculation, $S_{-j}$ is the complementary action subspace excluding $S_j$ from $S$, and $r_j$ is designed to take values between 0 and 1.

***Step 7:*** Select $S_j^* \in S$ to *maximize* the utility $r_j(S_j, S_{-j})$, given others' actions $S_{-j}$. Announce its decision $S_j^*$ to others, so that other sensors can update. Then go to ***Step 6***. For ***Step 6*** and ***Step 7***, a simplest illustrative procedure can be obtained by considering only one sensor and one target each time.

***Step 8:*** Sensor assignment ends.

## 2.2 Sensor Scheduling

For any assigned sensor-target pair, sensor scheduling will provide plans for specific sensor actions over a time horizon *H*. Unlike approaches in [15] and [16], which applied Q-learning [6] or one step look-ahead search strategy, we apply game theory and Markov decision process (MDP) [15] to deal with smart targets and search for the best sensing strategy with time horizon *H* ≥ 1. Assume that the sensor has different sensing modes, such as moving target indicator (MTI), high range resolution (HRR) and/or any other available modes. Similarly, assume that targets have more than one mode which can cause different measurement covariance. Both sensor and target might be able to move along different directions (if mobile), subjecting to some terrain constraints or other requirements.

For fully rational targets[3], our approach will apply game theory to generate action plans. The payoff function which will be maximized by the sensor is given by

$$\Psi_k^s = \sum_{n=k}^{k+H-1} \beta^{n-1} \Omega^s(n) \qquad (7)$$

[2] They can also be specific target entities or individual search areas.

[3] Here we assume that those rational targets will always choose the best action obtained using game theory assuming their opponents have complete knowledge of the game.

where $\beta \in [0,\ 1]$ is the time discount factor, $\Omega^s(n)$ is the payoff at the $n$th step. Here the payoff function includes the information gain related to the error covariance and discounted by the carefully normalized costs to be explained next. An issue in payoff function design is how to convert the numbers in different units into a comparable payoff function. A common practice is to select some normalizing factors/matrices and transform all terms into unitless quantities where specific techniques are used for different terms [4]. We use the payoff for sensor *s* at the *k*-th step given by

$$
\begin{aligned}
\Omega^s(k) = {} & \frac{1}{\det(P(S_{\text{mod}\,e}(k), T_{\text{mod}\,e}(k), S_{\text{move}}(k), T_{\text{move}}(k)))} \\
& - C_o^s(S_{\text{mod}\,e}(k), k) - \delta^s(k) C_c^s(k) \\
& - (\lambda^s)^{\Delta t_k^s} C_l^s(S_{\text{mod}\,e}(k), k) - C_m^s(S_{\text{mode}}(k), S_{\text{move}}(k), k) \\
& + C_o^t(T_{\text{mod}\,e}(k), k) + \delta^t(k) C_c^t(k) \\
& + (\lambda^t)^{\Delta t_k^t} C_l^t(T_{\text{mod}\,e}(k), k) + C_m^t(T_{\text{mode}}(k), T_{\text{move}}(k), k)
\end{aligned}
\tag{8}
$$

where $\det(P(S_{\text{mod}\,e}(k), T_{\text{mod}\,e}(k), S_{\text{move}}(k), T_{\text{move}}(k)))$ is the determinant of the a posterior covariance matrix when sensor $s$ is in $S_{\text{mod}\,e}(k)$ and target is $T_{\text{mod}\,e}(k)$. $C_o^s(S_{\text{mod}\,e}(k), k)$ is the sensor operation cost at time step $k$ in mode $S_{\text{mod}\,e}(k)$. In (8) $\delta^s(k)$ is defined as

$$
\delta^s(k) = \begin{cases} 1, & \text{if } k = 1 \\ 1, & \text{if } S_{\text{mod}\,e}(k) \neq S_{\text{mod}\,e}(k-1) \text{ and } k \geq 2 \\ 0, & \text{else} \end{cases}
\tag{9}
$$

and $C_c^s(k)$ is the cost related to changing mode; $C_l^s(S_{\text{mod}\,e}(k), k)$ is the cost related to taking one mode continuously for more than one time step; $\Delta t_k^s$ is the number of the time steps that sensor continuously takes $S_{\text{mod}\,e}(k)$; $\lambda^s$ is the corresponding base of the exponential function. The reason why we should consider such long-term cost is that under some situations, staying in one mode for too long does hurt. For example, a sensor might not be able to operate in one mode continuously since long time operation can cause overheating and thus the sensing accuracy can not be guaranteed. Similarly, a smart target might want to choose to hide itself when it detects that a sensor keeps tracking it. However, such "hide" mode might require the target to stay somewhere or move very slowly since the target must obey certain order such as "reach some as quick as possible". As a result, we assume that the longer it operates in one mode, the more marginal penalty it will undertake. For many situations such marginal penalty can be approximated by the exponentially increasing factor. For cases in which there are no such penalty, $\lambda^s$ can be set to 0 (no penalty) or 1 (not exponentially increasing). $C_m^s(S_{\text{mode}}(k), S_{\text{move}}(k)k)$ is the cost of movement, if the sensor is mobile. It is in $C_m^s(S_{\text{mode}}(k), S_{\text{move}}(k)k)$ term where the terrain information and constraints should be incorporated. For simple cases, such costs can be looked up from a table. Note that the meanings of $C_c^t(k)$, $C_l^t(T_{\text{mod}\,e}(k), k)$, $(\lambda^t)^{t_k^t} C_l^t(T_{\text{mod}\,e}(k), k)$, and $C_m^t(T_{\text{mode}}(k), T_{\text{move}}(k), k)$ can be explained symmetrically.

The payoff function which will be maximized by the target is defined as the negative of the sensor payoff:

$$
\Psi_k^t = -\Psi_k^s = -\sum_{n=k}^{k+H-1} \beta^{n-1} \Omega^s(n)
\tag{10}
$$

After setting up the payoff function, the self-stable Nash solution [11], which includes at which time the entity (sensor or target) should take which mode and move to where, can be calculated assuming both parties have the complete knowledge of the game (that is, perfect information structure). In our simulation study (section 3), we assumed perfect information structure for simplification.

Such game theoretic approach can provide self-enforcing solutions, which means when the sensor chooses the action corresponding to some appropriate equilibrium, if the target does not choose the corresponding action at the same equilibrium, the sensor will achieve higher payoff and the target will results in lower payoff. The disadvantage of this approach is its relatively heavier computational load. Usually when the target is treated as high-tech and high value opponent with powerful anti-detection and calculation equipments, such game theoretic approach will be suitable owing to the following facts. 1) Such high-tech targets have the capability to play the game and find out the same equilibrium. 2) Such high value targets usually dare not take risks.

Sometimes, smart targets will not always choose to behave according to the rational game theoretic solution. One reason is that they may not have adequate computational equipment/resource to play the game, thus they tend to choose some relatively simpler behavior patterns. Another reason is that they might want to break the expectations from time to time so that their behaviors look more unpredictable, thus gain in alternative aspects (such as inferences about their mission, identification about their classes and labels, etc.). Some research [15] models such smart targets by assuming that they would change mode or state with some predetermined transitional probability. However, we believe that such assumption might not model the evasive targets "smart and dynamic enough". From our perspective, for such smart targets, a more reasonable refined assumption to approximate their behavior pattern is as follows:

***Dynamic probabilistic model:*** *If a target has* $p_1 \in [0,\ 1]$ *confidence that it is being monitored, it will choose to take an action (if possible) that causes more tracking difficulties with* $p_2 \in [0,\ 1]$ *probability. In* *addition, $p_2$ is positively correlated to $p_1$. That is to say, when $p_1$ increases, $p_2$ increases as well. Sometimes one can have $p_1 = p_2$.*

We refer to the above refined model as *dynamic probabilistic model*. Note that $p_1$ and $p_2$ are not predetermined and might change greatly according to the dynamic situations. One can see that targets in our model appear smarter and more dynamic, and thus more difficult to track. In addition, our assumption naturally fits two extreme conditions which can not be easily accommodated by existing methods: 1) when the target has absolutely no intelligence, we only need to set $p_2$ as 0; 2) when the target has high intelligence and full rationality, we only need to set $p_1 = p_2 = 1$ so that the problem becomes one step look-ahead game with much less computational cost. In either extreme case, the optimization procedure remains the same. Moreover, the above model is still simple and easy to implement, which makes it more attractive to ordinary smart targets. As a result, this method will also be suitable to analyze most situations with low-tech or low value smart targets.

The following analysis is based on the new target behavior model. Clearly, a sensor should choose the strategy that can maximize the payoff which is calculated according to some predicted probability:

$$\Psi_k^{s,new} = \sum_{m1=1}^{\mathrm{MODE}_t} \sum_{m2=1}^{MOVE_t} \left( q_k^t(m1,m2) \sum_{n=k}^{k+H-1} \beta^{n-1} \Omega^s(n) \right) \quad (11)$$

where $\mathrm{MODE}_t$ is the number of the target modes, ; $MOVE_t$ is the number of target move; $q_k^t(m1,m2)$ is the probability that the target will choose $\mathrm{MODE}_t$ and $MOVE_t$ at time step $k$. The definition of $\Omega^s(\cdot)$ is same as in the game theoretic approach. Note that the $T_{\mathrm{mod}e}(k)$ in the definition of $\Omega^s(\cdot)$ corresponds to $m1$. All other definitions follow the notations used before.

In practical applications, when it is difficult to determine which kinds of targets they are, mature learning mechanisms such as the fictitious play [17] need to be applied to help make a decision.

# 3 Simulation study

We implemented a prototype of the proposed HSM sensor management scheme. For simplicity, we emphasize a 10 sensor example, with a predefined sensor assignment case, with a two mode system. HSM first performs sensor assignment, then performs sensor scheduling based on the assigned sensor-target pairs

The output of sensor assignment module is a matrix *A* with non-negative integers. The columns are for targets. The rows are for sensors. Each element, $a_{ij}$, of the matrix *A* indicates how many channels of sensor *i* have been assigned to target *j*. As a result, $a_{ij}$ is nonnegative and no larger than the maximum channel capacity of the sensor.

A typical solution to sensor assignment problem for a single-channel is shown in Table 1 where each sensor has only one channel.

**Table 1 Single channel sensor assignment**

| | T 1 | T 2 | T 3 | T 4 | T 5 |
|---|---|---|---|---|---|
| S 1 | 0 | 0 | 1 | 0 | 0 |
| S 2 | 0 | 0 | 0 | 0 | 1 |
| S 3 | 0 | 1 | 0 | 0 | 0 |
| S 4 | 1 | 0 | 0 | 0 | 0 |
| S 5 | 0 | 0 | 0 | 1 | 0 |
| S 6 | 0 | 0 | 0 | 0 | 1 |
| S 7 | 1 | 0 | 0 | 0 | 0 |
| S 8 | 1 | 0 | 0 | 0 | 0 |
| S 9 | 1 | 0 | 0 | 0 | 0 |
| S 10 | 1 | 0 | 0 | 0 | 0 |

A typical solution for a multi-channel sensor assignment problem is shown in Table 2 where each sensor can have more than one channel. In this simulation study, the numbers of sensor channels are randomly generated.

**Table 2 Multiple channel sensor assignment**

| | T1 | T 2 | T 3 | T 4 | T 5 |
|---|---|---|---|---|---|
| S 1 | 1 | 1 | 0 | 0 | 1 |
| S 2 | 0 | 0 | 2 | 0 | 1 |
| S 3 | 4 | 0 | 0 | 0 | 0 |
| S 4 | 0 | 1 | 0 | 0 | 0 |
| S 5 | 0 | 5 | 0 | 0 | 0 |
| S 6 | 0 | 2 | 0 | 0 | 0 |
| S 7 | 1 | 0 | 1 | 3 | 0 |
| S 8 | 4 | 0 | 0 | 0 | 0 |
| S 9 | 1 | 0 | 0 | 0 | 0 |
| S10 | 4 | 0 | 0 | 1 | 0 |

In simulations with the implemented prototype for sensor scheduling module, we assume a sensor has two modes. Mode 1 is a mode that provides better covariance but is easily being detected by smart target (for example, the sensor emits strong signals – active mode). Mode 2 is just the opposite. For example, the sensor is operating on the passive mode. Similarly, a target is assumed to be "smart" (the pure game case or the dynamic probabilistic model) and also has two modes. Mode 1 is easy to be tracked but easy to operate. Mode 2 is more like a "hide" mode which is difficult to be tracked but more expensive to operate or persist. Entities are assumed to be able to move in 3D spaces with different cost of movement related to the motion toward different direction.

Table 3 and Table 4 are for the pure game situation, with look ahead horizon *H*=2.

**Table 3: Game sensor mode scheduling with *H*=2**

| Timestep index | Step 1 | Step 2 |
|---|---|---|
| Sensor mode | 2 | 1 |
| Target mode | 1 | 1 |

**Table 4: Game sensor mode scheduling with *H*=2**

| Timestep index | Step 1 | Step 2 |
|---|---|---|
| Sensor move | Move down | Move down |
| Target move | Move forward | Move forward |

We can see that to induce the target to keep mode 1, the sensor can first take mode 2 for 1 time step before it really choose mode 1, which can provide better covariance. In the sensor move planning results (Table 4), the different motions for sensor and target are partly due to simple practical terrain constraints: for an airborne sensor, to move down is relatively easier and better for achieving higher accuracy. However, for a ground target, to move forward is often the best choice for easiness and for completing missions. In the future terrain settings can be expanded to accommodate more complex geological information. An *H*=3 simulation is as follows (Table 5). Analysis is similar to *H*=2 case.

**Table 5: Game sensor mode scheduling with *H*=3**

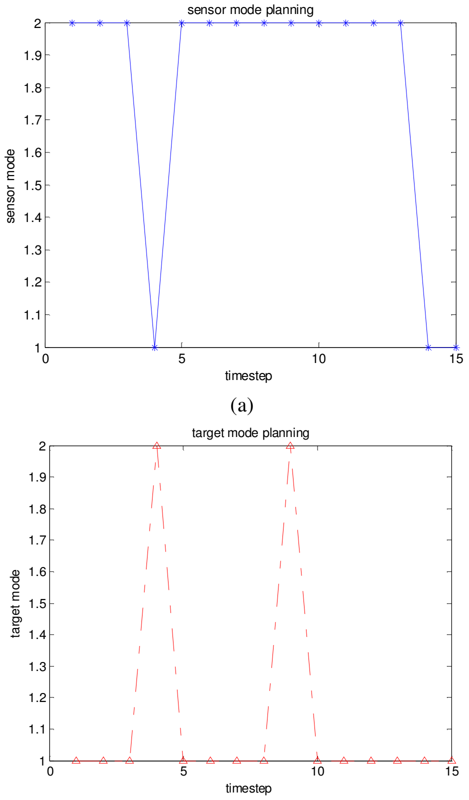
| Timestep index | Step 1 | Step 2 | Step 3 |
|---|---|---|---|
| Sensor mode | 2 | 2 | 1 |
| Target mode | 1 | 1 | 1 |

Note that if we do not consider the long time penalty in the payoff function, game theory would tend to recommend the sensor to take mode 1 and the target to take mode 2. This is reasonable and can be analyzed similar to prisoner's dilemma [18]: if no other penalty, for a sensor, no matter which mode the target takes, to choose mode 1 is always the best choice. Similarly we can find that mode 2 is always the best choice for target. This is confirmed in the following simulation plot (Table 6):

**Table 6: Game sensor mode scheduling with *H*=3**

| Timestep index | Step 1 | Step 2 | Step 3 |
|---|---|---|---|
| Sensor mode | 1 | 1 | 1 |
| Target mode | 2 | 2 | 2 |

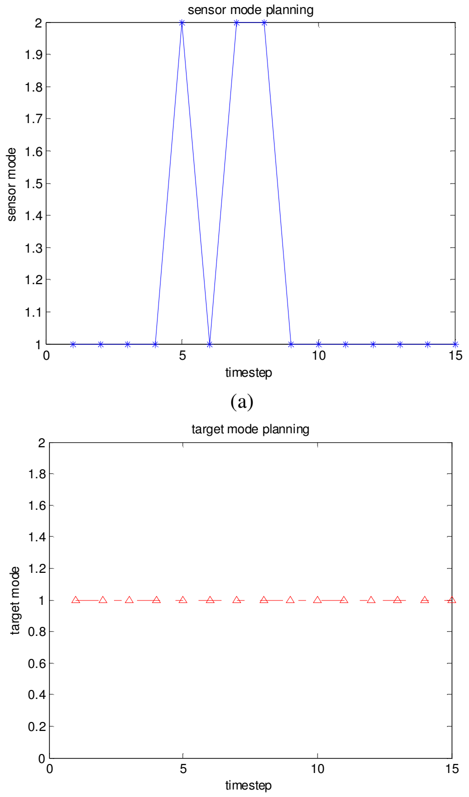
Figure 2 and Figure 3 are for the dynamic probabilistic model cases (*H*=15). Figure 2(a) and Figure 2(b) are for a case in which when the sensor is in mode 1, at each timestep the target will have probability *p*=0.2 to know whether it is tracked. In Figure 3(a) and Figure 3(b), such probability is 0.01.



(a)

(b)

**Figure 2:** Probabilistically modeled sensor mode scheduling with *p*=0.2



(a)

(b)

**Figure 3:** Probabilistically modeled sensor mode scheduling with $p$=0.01

We can see that for the “probability 0.2” case, since the target has higher probability being tracked, the sensor tends to operate on mode 2 (passive mode) for more time steps. The target also chooses “hide mode” (mode 2) for some time steps. For the “probability 0.01” case, the sensor tends to operate on mode 1 (strong signal model) for more time steps, since the target has very small chance to know whether it has been tracked.

## 4 Discussion and Conclusion

A game theoretic multiple mobile sensor management approach was proposed. Utilizing the developments from both information-theoretical and covariance-based sensor management approaches, we have formulated a scenario to track and identifying “intelligent” targets (ones that alter their behavior to signals detection). This approach can track smart targets under possibly adversarial environments. Covariance control and information theoretic sensor assignment were combined in a coherent manner where targets were modeled as entities with different levels of intelligence. Simulations illustrate the applicability of this approach.

Future work will be focused on incorporating a more general analysis of meaningful performance metrics, computational requirements, and joint control and estimation. In addition, incorporating a variety of tracking methods (e.g., Multiple Hypothesis Tracker, Interacting Multiple model, Joint Probabilistic Data Association Filter) as well as identification algorithms (PCA, Mutual Information, etc) with the hierarchical sensor management for smart targets will allow us to gain insights from empirical performance bounds on temporal/spatial/spectral tradeoffs as well as theoretical bounds (e.g., Cramer-Rao lower bound).

## 5 Acknowledgements

This research was supported in part by the Air Force under contract number FA8650-07-M-1161. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of the Air Force.

## References

[1] M. K. Kalandros, L. Trailovic, L. Y. Pao, and Y. Bar Shalom, “Tutorial on Multisensor Management and Fusion Algorithms for Target Tracking,” *Proceeding of the 2004 American Control Conference, Boston, Massachusetts*, June 30-July 2, 2004.

[2] J. Nash, “Optimal Allocation of Tracking Resources”, *Proceeding of IEEE Conference on Decision and Control*, New Orleans, LA, pp. 1177-1 180, 1977.

[3] W. Schmaedeke, “Information-based Sensor Management,” *Proc. of SPIE.*, 1955. 1993.

[4] M. Kalandros and L. Y. Pao, “Covariance Control for Multisensor Systems,” *IEEE Transactions on Aerospace and Electronic Systems*, Vol. 38, 2002.

[5] D. P. Bertsekas, Dynamic Programming and Optimal Control, Vols. 1 & 2, 2nd ed. Athena Scientific, 2000.

[6] Y. Li, L. W. Krakow, E. K. P. Chong, and K. N. Groom, “Dynamic Sensor Management for Multisensor Multitarget Tracking,” *2006 40th Annual Conference on Information Sciences and Systems*, Princeton, NJ, 22-24 March 2006, pp.1397-1402.

[7] A. Doucet, N. de Freitas and N. Gordon, Sequential Monte Carlo Methods in Practice, Springer, 2001.

[8] D. H. Nguyen, J. H. Kay, B. J. Orchard, and R. H. Whiting, “Classification and Tracking of Moving Ground Vehicles,” *Lincoln Laboratory Journal*, Vol 13, No. 2, 2002.

[9] T. Vercauteren, D. Guo, and X. Wang, “Joint Multiple Target Tracking and Classification in Collaborative Sensor Networks,” *IEEE Journal on Selected Areas in Communications*, Vol. 23, No. 4, April 2005.

[10] J. Dowdall, I. T. Pavlidis, and P. Tsiamyrtzis, “Coalitional Tracking in Facial Inferred Imaging and Beyond,” *Proc. of the 2006 Conference on Computer Vision and Pattern Recognition Workshop*, 2006.

[11] C. Camerer, *Behavioral Game Theory*, Princeton University Press, 2003.

[12] G. Bendall and F. Margot, Greedy Type Resistance of Combinatorial Problems, *Discrete Optimization*, pp.288-298, 2006.

[13] M. Irani and P. Anandan, “Robust MultiSensor Image Alignment,” Available at http://www.wisdom.weizm ann.ac.il/~irani/PAPERS/multiSensorAlignment.pdf.

[14] W. Schmaedeke and K. Kastella. “Information Based Sensor Management and IMMKF: *Proc. SPIE Conf Signal Processing of Small Targets*, Vol. 3373, Orlando. FL. pp. 390-401. 1998.

[15] C. Kreucher, D. Blatt, A. Hero, and K. Kastella, “Adaptive Multi-modality Sensor Scheduling for Detection and Tracking of Smart Targets,” *Digital Signal Processing*, Elsevier, 2005.

[16] K. C. Chang, “Level I and Level II Target Valuations for Sensor management,” *Journal of Advances in Information Fusion*, Vol. 1, No. 2, Dec. 2006.

[17] D. Fudenberg, and D. K. Levine, *The Theory of Learning in Games*. Cambridge: MIT Press, 1997.

[18] A. Tversky, *Preference, Belief, and Similarity*, Selected Writings. MIT Press, 2004.